Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A method of operating a network processor, the method comprising:

writing, to a shared memory accessible by multiple packet processing engines, a dynamic packet rule set, each rule specifying a packet offset, a data pattern, and an action code;

writing, to an instruction store for the packet processing engines, execution instructions referencing the dynamic packet rule set; and

on at least one of the packet processing engines, while processing a packet and in response to the execution instructions, loading a first packet rule from the dynamic packet rule set, comparing packet data at the packet offset specified in the first packet rule to the data pattern specified in the first packet rule, and, when the comparison indicates a match, performing an action indicated by the action code specified in the first packet rule,

wherein performing the action comprises incrementing a counter specified in the first packet rule, wherein the counter is located in a local memory area accessible by each packet processing engine, and wherein incrementing the counter comprises blocking other processors from accessing the counter during the increment.

2. (original) The method of claim 1, wherein the data pattern in each rule comprises a mask and a bit pattern, and wherein comparing packet data to the data pattern comprises masking the packet data using the mask, and comparing the masked packet data to the bit pattern.

3. (original) The method of claim 2, wherein the shared memory comprises a content-addressable memory (CAM), wherein the data pattern for at least some rules are stored in the CAM, and wherein masking the packet data and comparing the masked packet data to the bit pattern are performed by the CAM for data patterns stored in the CAM.

4. (cancelled)

5. (cancelled)

6. (original) The method of claim 1, wherein performing the action comprises loading a second packet rule specified in the first packet rule.

7. (original) The method of claim 1, further comprising, while processing the packet, loading additional packet rules from the dynamic packet rule set, and repeating the processing performed for the first packet rule for each of the additional rules.

8. (original) The method of claim 1, wherein each rule in the dynamic packet rule set comprises a rule valid field, the method further comprising checking the rule valid field prior to performing the action.

9. (original) The method of claim 1, wherein writing the dynamic packet rule set comprises arranging the rules in the set in order based on packet offset, with the rule having the smallest packet offset appearing first in the set.

10. (currently amended) An integrated circuit comprising:

a local memory to store a rule table, the rule table organized with entries comprising a packet offset, a data pattern, and an action;

a packet data queue to receive packet data;

a rule fetch unit to fetch rules from the rule table;

a packet data fetch unit to fetch a segment of packet data from the packet data queue, based on the packet offset fetched by the rule fetch unit;

match circuitry to compare the packet data segment fetched by the packet data fetch unit with the data pattern fetched by the rule fetch unit; [[and]]

an action unit to perform the action fetched by the rule fetch unit when the match circuitry indicates a match between the compared packet data segment and data pattern; and

the local memory also to store a counter table, wherein the action in a rule table entry comprises an index into the counter table, the action unit responding to the counter table index and a match indicated by the match circuitry by incrementing an entry in the counter table referenced by the counter table index.

11. (original) The integrated circuit of claim 10, wherein the data pattern in each rule table entry comprises a mask and a bit pattern, the match circuitry comprising:

mask circuitry to apply the mask in the fetched data pattern to the fetched packet data segment; and

compare circuitry to compare the masked packet data segment to the bit pattern in the fetched data pattern.

12. (cancelled)

13. (original) The integrated circuit of claim 10, wherein the action in a rule table entry comprises an index to a second rule table entry, the action unit responding to the rule table entry index and a match indicated by the match circuitry by setting a success flag in the second rule table entry.

14. (original) The integrated circuit of claim 10, wherein each rule table entry comprises a rule valid field, the rule fetch unit reading the rule valid field on a fetched rule to determine whether to process the rule.

15. (original) The integrated circuit of claim 14, wherein one possible value of the rule valid field indicates an end of the rule table, the rule fetch unit resetting to the head of the rule table after fetching a rule wherein the rule valid field indicates the end of the rule table.

16. (original) A method of gathering statistics on packets received by a network processor, the method comprising:

configuring a core processor to dynamically accept packet rule requests and place corresponding packet rules in a packet rule set area at a first memory region in an addressable memory space, at least one rule in the packet rule set specifying a packet offset, a data pattern, and a counter offset;

configuring a set of packet processing engines to sequence through the packet rule set, retrieving one of the packet rules from the first memory region and comparing packet data from a received packet, at the offset specified in the retrieved packet rule, to the data pattern specified in the retrieved packet rule, and, when the comparison evaluates true, incrementing a counter, at the counter offset specified in the retrieved packet rule, within a second memory region in the addressable memory space; and

configuring the core processor to retrieve statistics from the counters in the second memory region.

17. (previously presented) The method of claim 16, wherein configuring the core processor to dynamically accept packet rule requests comprises configuring the core processor to allocate a counter offset in response to a packet rule request.

18. (original) The method of claim 16, further comprising the core processor arranging the rules in the packet rule set area in an order based on packet offset, with the rule having the smallest packet offset appearing first in the set.

19. (original) The method of claim 16, wherein configuring the core processor to retrieve statistics comprises loading a control plane process allowing a remote administrator to request and receive periodic statistics reports from the core processor for the rules in the packet rule set.

20. (original) The method of claim 16, wherein each rule has a fixed-length data pattern, the method further comprising configuring the core processor to pad a requested data pattern in a packet rule request up to the fixed length when the requested data pattern is shorter than the fixed length, and mask the padded portion of the data pattern.

21. (original) The method of claim 16, wherein each rule has a fixed-length data pattern, the method further comprising:

configuring the core processor to divide a requested data pattern in a packet rule request into multiple data patterns no larger than the fixed length when the requested data pattern is longer than the fixed length, and place the multiple data patterns in corresponding

multiple packet rules, the first of the multiple packet rules referencing the second or vice-versa; and

configuring the set of packet processing engines to fully process the second of the multiple packet rules only when processing the first of the multiple packet rules results in a comparison evaluating true.

22. (original) An article of manufacture comprising computer-readable media containing instructions that, when executed by a network processor, cause that network processor to perform a method comprising:

dynamically accepting packet rule requests at a core processor and placing corresponding packet rules in a packet rule set area at a first memory region in an addressable memory space, at least one rule in the packet rule set specifying a packet offset, a data pattern, and a counter offset;

sequencing through the packet rule set with a set of packet processing engines, retrieving one of the packet rules from the first memory region and comparing packet data from a received packet, at the offset specified in the retrieved packet rule, to the data pattern specified in the retrieved packet rule, and, when the comparison evaluates true, incrementing a counter, at the counter offset specified in the retrieved packet rule, within a second memory region in the addressable memory space; and

the core processor retrieving statistics from the counters in the second memory region for distribution outside of the network processor.

23. (original) The article of manufacture of claim 22, wherein the method further comprises the core processor arranging the rules in the packet rule set area in an order based on packet offset, with the rule having the smallest packet offset appearing first in the set.

24. (original) The article of manufacture of claim 22, wherein the core processor retrieving statistics comprises communicating with a remote administrator, allowing the remote administrator to request and receive periodic statistics reports from the core processor for the rules in the packet rule set.

25. (original) The article of manufacture of claim 22, wherein each rule has a fixed-length data pattern, the method further comprising the core processor padding a

requested data pattern in a packet rule request up to the fixed length, and masking the padded portion of the padded data pattern, when the requested data pattern is shorter than the fixed length.

26. (original) The article of manufacture of claim 22, wherein each rule has a fixed-length data pattern, the method further comprising:

the core processor dividing a data pattern in a packet rule request into multiple data patterns no larger than the fixed length when the requested data pattern is shorter than the fixed length, and placing the multiple data patterns in corresponding multiple packet rules, the first of the multiple packet rules referencing the second or vice-versa; and

the set of packet processing engines fully processing the second of the multiple packet rules only when processing the first of the multiple packet rules results in a comparison evaluating true.

27-30. (cancelled)